Varun Jose Madanu

Albany, NY ∘ +1 518-596-4160 ∘ varunjosemadanu@gmail.com ∘ LinkedIn ∘ Portfolio

EDUCATION

University at Albany, SUNY — New York, USA
Master of Science, Computer Science — 08/2023 - 05/2025
Coursework: Algorithms and Data Structures, Software Engineering, Machine Learning, Database Systems

KPRIT Institute of Technology and Science — Hyderabad, India
Bachelors in Computer Science and Engineering — 06/2019 - 06/2023
Coursework: Data Science, OOPS using C++, Operating Systems, Network Security, Distributed Systems.

PROFESSIONAL EXPERIENCE

NYS Department of Transportation — 06/2024 - 05/2025
Data Analyst — Albany, NY, USA

- Utilized Python (Pandas, NumPy, OpenPyXL) and SQL for advanced data manipulation and querying, boosting report efficiency by 70%.
- Automated reporting workflows using Python, Power BI, SAP Crystal Reports, and Jupyter Notebooks, saving 6+ hours weekly.
- Developed interactive dashboards with Tableau, integrating data from SQL Server and Python pipelines, reducing decision-making time by 3+ hours/week.
- Conducted data validation and cleaning using SQL, Python, and Excel VBA, improving data integrity and reporting accuracy.
- Reduced manual tasks by 30% through automation and ETL optimization using Apache Airflow and Power Query.

Cognier Insights — 09/2022 - 06/2023
Full Stack Developer — Hyderabad, India

- Built a scalable retail app using ASP.NET Core and Angular 10, resulting in 30% increased customer satisfaction.
- Deployed system with Azure API Gateway and RabbitMQ for asynchronous messaging.
- Implemented Entity Framework Core with SQL Server for dynamic data processing.
- Created responsive UI with HTML, CSS, JavaScript, and Bootstrap.
- Conducted unit testing using XUnit and end-to-end testing using Karma and Jasmine.
- Utilized OAuth2, JWT, and SSL for secure transactions and role-based access control.
- Managed code with Git and collaborated using JIRA in Agile sprints.

TECHNICAL SKILLS

Programming Languages: C#, Java, Python, SQL, JavaScript, HTML, CSS, C++
.NET Technologies: ASP.NET Core Web API, ASP.NET MVC, ADO.NET, Entity Framework, Entity Framework Core
Data Analytical Tools: Tableau, Google Sheets, R Programming (via RStudio), Google Data Studio (Looker Studio), Power BI, SQL, Python (NumPy, SciPy, Statsmodels, scikit-learn, Pandas), SAS
Web Technologies: Angular 10+, AngularJS, TypeScript, HTML, CSS, Bootstrap, jQuery, JavaScript, AJAX
Databases & Cloud Technologies: SQL Server, PostgreSQL, MongoDB, Azure, AWS, GCP
Web Servers and Services: IIS, Apache Tomcat, RESTful, SOAP
Version Control Tools: Git, SVN, TFS
Methodologies: Agile, Waterfall

PROJECT EXPERIENCE

FinanceMate | ngular, JavaScript, Node.js, Express.js, PostgreSQL, RESTful APIs, Google Cloud Platform (GCP), App Engine, Cloud SQL, Terraform, JWT, Docker, GitHub Actions, Cloud Build, Nginx. 🔗

- I built a full-stack web app using Angular and JavaScript for the frontend and Node.js with Express.js for the backend. Data was stored in an RDBMS (PostgreSQL) and accessed via custom RESTful APIs. The application was deployed on Google Cloud Platform (GCP) using App Engine and Cloud SQL, with infrastructure managed via Terraform.

MobileRecommender | React.js, Node.js, Express.js, PostgreSQL, RESTful APIs, AWS (EC2, RDS, S3), Terraform, JWT, Docker, GitHub Actions, AWS CodeDeploy, Nginx. 🔗

- I developed a user-friendly mobile recommender system using React.js for the frontend and Node.js with Express.js for backend services. The system leverages RESTful APIs to serve personalized recommendations from a structured PostgreSQL database. It is hosted on Amazon Web Services (AWS) using EC2 for compute, RDS for database hosting, and S3 for static assets.

Prediction Model for Obstructive Sleep Apnea from Facial Depth Maps | Python, CNN, TensorFlow Keras, VGG-19, OpenCV, NumPy, Pandas, Pickle, Matplotlib, Tkinter, JSON. 🔗

- Developed a desktop app to predict Obstructive Sleep Apnea (OSA) using facial depth maps with VGG-19 and TensorFlow Keras. Used OpenCV and NumPy for image preprocessing, and Pandas and Pickle for data handling and model persistence. GUI was built with Tkinter, and Matplotlib was used for performance visualization. The app supports dataset upload, model training, prediction, and result display.